



DECUS

PROGRAM LIBRARY

DECUS NO.	6/10-43a
TITLE	PDP-8 SIMULATOR ON THE PDP-10
AUTHOR	D. McClure
COMPANY	Digital Equipment Corporation Waltham, Massachusetts
DATE	September 5, 1969
SOURCE LANGUAGE	MACRO-10

ATTENTION

This is a USER program. Other than requiring that it conform to submittal and review standards, no quality control has been imposed upon this program by DECUS.

The DECUS Program Library is a clearing house only; it does not generate or test programs. No warranty, express or implied, is made by the contributor, Digital Equipment Computer Users Society or Digital Equipment Corporation as to the accuracy or functioning of the program or related material, and no responsibility is assumed by these parties in connection therewith.

DEC 12

PROGRAM 1100 AM



11:11 AM

11:11 AM

11:11 AM

11:11 AM

11:11 AM

11:11 AM

11:11 AM

11:11 AM

11:11 AM

11:11 AM

WHAT IS A PDP-8 SIMULATOR?

THE SIMULATOR ATTEMPTS TO DUPLICATE ALL PDP8 FUNCTIONS (INCLUDING IO DEVICE TIMINGS) SO THAT THE RESULTS OF STARTING ANY PROGRAM IN EITHER THE SIMULATOR OR AN ACTUAL PDP8 WILL BE IDENTICAL.

TO DO THIS WE HAVE SIMULATED CORE WHICH CONTAINS PDP8 12BIT WORDS WHICH ARE THE PDP8 PROGRAM. THESE "PDP8 WORDS" ARE THEN READ, DECODED, AND EXECUTED AS PDP8 INSTRUCTIONS. UNDER FAVOURABLE CONDITIONS THE SIMULATION PROCEEDS AT A RATE APPROXIMATELY 1/30 THE RATE AN ACTUAL PDP8 COULD EXECUTE THE CODE.

THE FOLLOWING DEVICES MAY BE SIMULATED:
CONSOL TELETYPE KEYBOARD AND TELEPRINTER
HIGH SPEED PAPER TAPE AND PUNCH
CALCOMP PLOTTER
DDF32(MINI) DISK

USING THE PDP8 SIMULATOR

FOR OUR DISCUSSION WE WILL ASSUME THE READER IS FAMILIAR WITH THE PROGRAMMING AND CONSOLE OPERATION OF THE PDP8. FURTHER SOME KNOWLEDGE OF PDP-10 COMMANDS IS ASSUMED. BE SURE TO READ THE SECTION TTY.MAC IN THIS FILE BEFORE ACTUALLY RUNNING THE SIMULATOR. TO RUN THE SIMULATOR WE TYPE:

"RUN DTANIDEMO"

THE SIMULATOR WILL IDENTIFY ITSELF AND INDICATE ITS READINESS TO ACCEPT COMMANDS BY TYPING AN *. THE COMMANDS WHICH ARE CURRENTLY IMPLEMENTED ARE:

BINARY	:RUN THE BINARY LOADER FROM THE HIGH SPEED PAPER TAPE READER. SEE THE EXAMPLE.
BOOTSTRAP	:LOAD THE FIRST BLOCK FROM THE DF32 INTO THE HIGH PAGE OF CORE. (USED TO LOAD THE DISK MONITOR IF IT IS ON THE DISK)
C	:EQUIVALENT TO DEPRESSING THE CONTINUE SWITCH ON THE CONSOLE OF A PDP8.
D N1:N2,N3	:DEPOSIT THE 12BIT NUMBER N2 IN LOCATION N1 AND THE NUMBER N3 IN LOCATION N1+1.
D N1:N2,N3:N4	:DEPOSIT N2 IN LOCATION N1 AND DEPOSIT N4 IN LOCATION N3.
DDT	:EXIT TO DDT IF DDT WAS LOADED WITH THE SIMULATOR. TO RETURN CONTROL TO THE SIMULATOR TYPE OPR\$G.
DISPLAY	:DISPLAY THE STATE OF THE MAJOR REGISTERS AND THE FLAGS. SEE THE EXAMPLES.
DP	:EQUIVALENT TO DEPRESSING THE DEPOSIT SWITCH ON THE CONSOLE OF A PDP8.
E	:EQUIVALENT TO DEPRESSING THE EXAMINE SWITCH ON THE CONSOLE OF A PDP8.

EX N1,N2-N3	:TYPE OUT THE CONTENTS OF THE LOCATIONS N1, AND LOCATIONS N2 THROUGH N3.
DUMP	:DUMP ALL OF THE PDP8'S CORE LOCATIONS ON THE LINE PRINTER FILE AS AN OCTAL DUMP.
DUMP N1,N2-N3	:DUMP THE CONTENTS OF CORE LOCATION N1 AND LOCATIONS N2 THROUGH N3 AS AN OCTAL DUMP ON THE LINE PRINTER FILE.
EXIT	:CLOSES ALL OPENED FILES. USED IF CHANGING FILES ON SIMULATED DEVICES (E.G. BINARY LOADING AN OVERLAY TO A PROGRAM WITH A SECOND TAPE)
FILES	:STARTS A DIALOGUE WHICH PERMITS CHANGING FILE NAMES AND LOGICAL DEVICES. SEE THE EXAMPLES,
INTERRUPT &N	:WILL SET THE DRIVER TO INTERRUPT WHEN THE SIMULATED RUNNING TIME HAS INCREASED N MICROSECONDS.
INTERRUPT N	:SET THE DRIVER TO INTERRUPT WHEN THE RUNNING TIME REACHES N.
L	:EQUIVALENT TO DEPRESSING THE LOAD-ADDRESS SWITCH ON THE CONSOLE OF A PDP8.
SR N	IN IS AN OCTAL NUMBER SIX DIGITS LONG. LOAD THE DATA FIELD SWITCHES WITH THE FIRST OCTAL DIGIT, LOAD THE INSTRUCTION FIELD SWITCHES WITH THE SECOND OCTAL DIGIT, AND LOAD THE DATA SWITCHES WITH THE FOUR LOW ORDER OCTAL DIGITS.
PC N	IN IS AN OCTAL NUMBER SIX DIGITS LONG. LOAD THE DATA FIELD SWITCHES WITH THE FIRST OCTAL DIGIT, LOAD THE INSTRUCTION FIELD REGISTER WITH THE 2ND DIGIT, LOAD THE ADDRESS REGISTER WITH THE FOUR LOW ORDER OCTAL DIGITS.
ST N	:START THE SIMULATED PDP8 WITH THE ADDRESS REGISTERS INITIALIZED WITH N.
S	:EQUIVALENT TO DEPRESSING THE START SWITCH ON THE CONSOLE OF A PDP8.

SS :SINGLE-STEP. THE SIMULATOR WILL EXECUTE ONE PDP8 INSTRUCTION AND THEN HALT SHOWING ITS REGISTERS.

ZIP :STARTS A DIALOGUE WHICH CHANGES THE SIMULATED SPEED OF SOME PERIPHERALS. SEE EXAMPLES.

RIM :RUN THE RIM LOADER FROM THE HIGH SPEED PAPER TAPE READER.

NOTE: WHEN A SPECIAL CHARACTER SUCH AS - : , . FOLLOWS A NUMBER, IT MUST FOLLOW IMMEDIATELY AFTER THE NUMBER WITH NO INTERVENING BLANKS.

NOTE: WHEN EVER USING PDP-10 PIP TO TRANSFER FILES BETWEEN DEVICES FOR USE BY THE SIMULATOR, IT IS GENERALLY GOOD PRACTICE TO USE THE /I (IMAGE MODE) SWITCH.

NOTE: IF THE TELETYPE SEEMS TO GO DEAD WHEN YOU STOP THE SIMULATOR TYPE CONTROL S AND CONTROL B TO RETURN TO NORMAL OPERATION.

SAMPLES AND EXAMPLES

*C
RUN DTAN:DEMO.SAV

YOU HAVE A 4K SIMULATED PDP8(INLINE VERSION) WITH AN UNPACKED MEMORY.

PLOTTER

*EX 7756
7756 6014

*EX 7760-7763
7760 5357
7761 6016
7762 7106
7763 7006

*RIM
END OF FILE ON PTR
*DISPLAY
PC 7762
AC 0377
IR 0006
MA 7761
MB 6016
SR 0 0 0000

SIMULATED RUNTIME 3,034,221,75 SECONDS

RUNNING TIME 84,600 SECONDS

SINCE SOME PROGRAMS MAY HAVE LOOPS SUCH AS:

```
RSF      /READER READY?
JMP      .-1    /NO SO LOOP BACK AND TEST AGAIN
```

TO READ CHARACTERS, IT MAY BE DESIREABLE TO INCREASE THE APPARENT SPEED OF SOME OF THE PERIPHERAL DEVICES TO REDUCE THE WORK LOAD ON THE SIMULATOR. THIS IS ACCOMPLISHED USING THE ZIP COMMAND AS DEMONSTRATED IN THE FOLLOWING EXAMPLE:

```
*ZIP
READER SPEED WAS 454CPS CHANGE TO 1000
PUNCH SPEED WAS 77CPS CHANGE TO
KEYBOARD SPEED WAS 12CPS CHANGE TO
TELEPRINTER SPEED WAS 12 CPS CHANGE TO 500
PLOTTER SPEED WAS 454CPS CHANGE TO
```

IN THE ABOVE DIALOGUE WE CHANGED THE HIGH SPEED PAPER TAPE SPEED FROM 300 CHARACTERS PER SECOND TO 512. (NOTE AGAIN THAT MOST NUMBERS ACCEPTED OR TYPED BY THE DRIVER ARE OCTAL) THE SPEEDS OF THE PLOTTER AND HIGH SPEED PAPER TAPE PUNCH WERE NOT CHANGED. FROM READING ABOUT THE STRUCTURE OF THE SIMULATOR YOU KNOW THAT EVERY 1/10 OF A SECOND THE TELETYPE IS POLLED FOR INPUT. THIS POLLING IS A SIGNIFICANT OVERHEAD SO AVOID SPEEDING THE KEYBOARD UP UNNECESSARILY.

THE PDP-10 IS A FILE ORIENTED SYSTEM. THE SIMULATOR HAS TWO COMMANDS SPECIFICALLY RELATED TO THIS STRUCTURE, "EXIT" AND "FILES". THE FIRST IS USED TO CLOSE ANY OPENED FILES. THE SECOND IS USED TO SELECT THE DEVICES AND FILE NAMES FOR INPUT AND OUTPUT BY THE VARIOUS SIMULATED PERIPHERALS. AN EXAMPLE OF THE "FILES" COMMAND FOLLOWS:

```
*FILES
READER PTR:PTR,RIM MODIFY THIS WITH
PUNCH PTP:PTP,OUT MODIFY THIS WITH DSK:PUNCH.FIL
PRINTER LPT:FILE,LST MODIFY THIS WITH DSK:
PLOTTER PLT:PLOTS.GPH MODIFY THIS WITH .EXT
DF32 DSK:DF32.FIL MODIFY THIS WITH DISK2
```

IN THE ABOVE EXAMPLE WE:

1. LEFT THE PAPERTAPE READER FILE DEVICE AND FILE NAME UNCHANGED
2. CHANGED THE PUNCH FILE NAME AND EXTENSION TO PUNCH.FIL, AND CHANGED IT TO OUTPUT TO THE DISK,
3. CHANGED THE LINE PRINTER FILE TO THE DISK, BUT LEFT THE FILE NAME AND EXTENSION THE SAME,
4. CHANGED THE EXTENSION OF THE PLOTTER FILE TO "EXT"
5. CHANGED THE "MINI" DISK FILE NAME TO DISK2.

NOW LET'S RUN A DEMONSTRATION. MOUNT THE DECTAPE FROM DECUS ON A DRIVE WITH LOGICAL NAME "FOO".(,ASSIGN DTA FOO) THEN TYPE:

1. ,RUN FOO:DEMO
2. WHEN THE SIMULATOR IS READY, CHANGE THE SPEED OF THE PAPER TAPE READER TO 1000000 CPS.
3. THEN CHANGE THE PAPER TAPE INPUT FILE NAME TO FOO:FOCAL.TPE. ALTHOUGH WE ARE USING THE DISK, ANY SIMULATED DEVICE (WITH THE EXCEPTION OF THE DF32 AND TELETYPE) COULD USE ANY PDP-10 DEVICE.
4. WHEN THE SIMULATOR IS READY, RUN THE BINARY LOADER.
5. WHEN THE SIMULATOR ANNOUNCES THAT THE PDP8 HAS HALTED SEE IF THE AC IS 0 (LINK WILL BE 1).
6. IF IT IS TYPE C TO CONTINUE RUNNING THE BINARY LOADER AND READ THE SECOND HALF OF THE BINARY 'FOCAL' TAPE INTO THE PDP8.
7. AGAIN SEE IF THE AC IS 0.
8. THEN START THE SIMULATOR AT 200.
9. 'FOCAL' WILL NOW RUN ON THE SIMULATED PDP8, AND START TYPING "CONGRATULATIONS"
10. YOU WILL QUICKLY NOTICE THAT THE TYPE OUT IS VERY SLOW,
11. SO TYPE "P.". THIS WILL RETURN CONTROL TO THE DRIVER.
12. NOW CHANGE THE SPEED OF THE TELEPRINTER TO 500CPS AND THEN CONTINUE RUNNING THE PROGRAM. 'FOCAL' IS NOW RUNNING!!

SECOND DEMONSTRATION

1. PIP DF32.FIL TO THE DISK WITH THE /I (IMAGE MODE) SWITCH.
2. ,RUN FOO:8KPDP8
3. SPEED THE TELETYPE TO 500CPS AND THE KEYBOARD TO 100 CPS.
4. BOOTSTRAP THE DISK MONITOR INTO CORE
5. RUN PIP TO GET A DIRECTORY OF THE DISK, I.E.,
 ,PIP
 *OPT-L
 *INP-S:

STRUCTURE OF THE PDP8 SIMULATOR

THE PDP8 SIMULATOR CONSISTS OF 14 MACRO SUBROUTINES AND 2 COMMON MACRO10 FILES(P,MAC AND F,MAC).

THE FILE PDP8,MAC CONTAINS THE BASIC SIMULATOR. IT KEEPS TRACK OF TIME, SETS SOME FLAGS, HANDLES THE PI SYSTEM AND DATA BREAK (RM08 ONLY) AND ALMOST ALL MEMORY REFERENCES. A TIME FLOW IS:

A.TIME CONTROL

- A. (CYCLE1)CHECK FOR FLAGS TO BE SET AND SPECIAL ROUTINES TO BE EXECUTED(KEYBOARD INPUT, DF32 DATA BREAK ETC.)
<APP. 6US IF NO FLAGS ARE TO BE SET>
- B. RM08 DATA BREAK HANDLING.
- C. CHECK FOR INTERRUPTS IF THE PI IS ON
<APP. 4US IF NO FLAGS ARE SET>

B.INSTRUCTION CYCLE

- A. (INSFET1)INSTRUCTION FETCH.
<APP. 10US>
- B. EFFECTIVE ADDRESS CALCULATION
<APP. 11US>
- C. INDIRECT ADDRESS CALCULATION
<APP. 10US>
- D. (DISP1)DISPATCH TO INDIVIDUAL INSTRUCTION ROUTINES(BY OP CODE)
<APP 4US>
- E. IF NOT AN IOT, THE INSTRUCTION IS EXECUTED IN PDP8,MAC. IF THE INSTRUCTION IS AN IOT A DISPATCH IS EXECUTED TO ITS OPCODE IN ITS DEVICE SUBROUTINE.
- F. RETURN TO TIME CONTROL FOR THE NEXT INSTRUCTION CYCLE.

THE OTHER SUBROUTINES ARE:

P,MAC PROVIDES THE DEFINITIONS OF SEVERAL CONDITIONAL ASSEMBLY PARAMETERS, TTCALL UUO'S, CERTAIN MACRO DEFINITIONS, ACCUMULATOR ASSIGNMENTS AND SOME BIT DEFINITIONS, SUCH AS THE BITS IN THE FLAGS,

F,MAC PROVIDES DEFINITIONS OF OTHER COMMON BITS AND WORDS ASSOCIATED WITH THE FILE STRUCTURE, SUCH AS BITS IN CFSTAT AND CDSTAT, AND WORD DEFINITIONS IN FILE DATA BLOCKS AND DEVICE DATA BLOCKS,

CORE,MAC THE SIMULATED CORE FOR THE PDP8. THE RIM AND BINARY LOADERS FOR THE PDP8 ARE ALWAYS ASSEMBLED IN THE HIGH PAGE OF CORE. IF THE CONDITIONAL ASSEMBLY PARAMETER PACK (IN P,MAC) IS A 0, THEN SIMULATED PDP8 WORDS ARE PACKED 3 WORDS TO A PDP10 WORD, OTHERWISE THEY ARE PLACED ONE PDP8 WORD TO ONE PDP10 WORD,

MEMORY,MAC CONTAINS THE MEMORY EXTENSION IOT'S. THE INSTRUCTION FIELD (IF), AND DATA FIELD (DFF) ARE WORDS IN THIS ROUTINE,

CONSOL,MAC CONTAINS THE FUNCTIONS OF MOST OF A PDP8'S CONSOL SWITCHES S(START),C(CONTINUE),L(LOAD ADDRESS),DP(DEPOSIT),E(EXAMINE),SS(SINGLE STEP)AND THE ROUTINES EXIT AND INI WHICH RESPECTIVELY CLOSE ALL FILES, AND INITIALIZE ALL FILES,

COMMON,MAC THE PAPER TAPE READER, PAPER TAPE PUNCH, PLOTTER, AND PRINTER USE PDP-10 DATA FILES SERVICED BY COMMON,MAC. ANY PDP-10 DEVICE MAY BE SUBSTITUTED FOR ANY OF THESE SIMULATED DEVICES PROVIDED ONLY THAT IT CAN HANDLE IO IN THE PROPER DIRECTION,

TTY.MAC HANDLES ALL IOT'S FOR THE TTY. THE KEYBOARD IS ALWAYS ACTIVE AND EVERY .1 SECONDS (ASSUMING 10CPS SPEED) INPUT IS ATTEMPTED VIA THE TTCALL UOJ'S. IF A CHARACTER WAS RECEIVED THE KEYBOARD FLAG IS SET (SEE P.MAC AND F.MAC FOR A DEFINITION OF FLAGS).

THE PDP-10 MONITOR INTERCEPTS CERTAIN SPECIAL CHARACTERS SO WE MUST RESORT TO SUBTERFUGE IF WE ARE TO BE ABLE TO TYPE THE FULL CHARACTER SET ON THE SIMULATED PDP8 TELETYPE. WE ACCOMPLISH THIS BY SAYING THAT:

*
*
*C
IS GIVEN TO THE PDP8 AS AN *
IS GIVEN TO THE PDP8 AS A RUBOUT
IS GIVEN TO THE PDP8 AS A CONTROL C
IN FACT A * FOLLOWED BY ANY ALPHABETIC CHARACTER WILL GIVE THE PDP8 THE CORRESPONDING CONTROL CHARACTER.
*
WILL IMMEDIATELY RETURN CONTROL TO THE DRIVER ROUTINE. (SINCE THE TELETYPE IS POLLED INFREQUENTLY THIS MAY NOT ACTUALLY HAPPEN QUICKLY.)

THE PDP-10 MONITOR FURTHER MEDDLES BY ECHOING CHARACTERS AND INSERTING LINEFEEDS AFTER CARRIAGE RETURNS. BUT AGAIN WE PERSEVERE BY HAVING THE SIMULATOR INITIATE HALF DUPLEX, AND TAPE MODES OF OPERATION JUST PRIOR TO RUNNING A PDP8 PROGRAM. IF YOU STOP THE SIMULATOR WHILE THE PDP8 WAS RUNNING, TYPING CONTROL B, CONTROL S WILL RETURN THE TELETYPE TO NORMAL OPERATION. (I HAVE ASSUMED THAT THE TELETYPE CONTROLLING THE JOB WILL BE FULL DUPLEX, IF IT ISN'T, YOU SHOULD MAKE A CHANGE IN THE DRIVER, AT OPR+2)

PTAPE.MAC HANDLES THE IOT'S FOR A HIGH SPEED PAPER TAPE READER AND A HIGH SPEED PAPER TAPE PUNCH. BECAUSE PAPER TAPE IS GENERALLY 8 CHANNEL, THE SIMULATION USES IMAGE MODE, THEREFORE TO PIP A PAPER TAPE TO THE DISK USE:

,R PIP
*DSK:FILE,EXT/I+PTRI

AND EQUIVALENTLY TO PIP A FILE FROM THE DISK TO PAPER TAPE.

LPTSER.MAC WAS INTENDED TO HANDLE LINE PRINTER IOT'S BUT DUE TO A LACK OF LINE PRINTER IOT SPEC'S WAS NEVER WRITTEN. IT DOES HANDLE THE FILE FOR THE DUMP COMMAND.

PLOTTER.MAC HANDLES THE IOT'S FOR AN INCREMENTAL PLOTTER.

DF32.MAC HANDLES ALL OF THE IOT'S FOR A SIMULATED DF32. THE VALUE OF DDF32(DEFINED IN P.MAC) IS THE NUMBER OF MINI DISKS IN THE SYSTEM. THE SIMULATION USES A PDP-10 DISK FILE TO SIMULATE THE STORAGE OF THE DISK. THREE 128IT WORDS ARE STORED IN ONE PDP10 WORD, HENCE EACH SIMULATED DF32 DISK TAKES UP APPROXIMATELY 256 PDP-10 DISK BLOCKS. THE SIMULATION MAY ALTERNATELY USE A DECTAPE FOR THE SIMULATED DISK STORAGE. BUT IF A DECTAPE IS USED, THE NORMAL PDP-10 FILE STRUCTURE IS NOT OBSERVED. HENCE THE FILE MAY NOT BE TRANSFERRED WITH PIP, AND WILL PROBABLY OVERWRITE ANY FILES WHICH WERE PREVIOUSLY ON THE DECTAPE. ONE DECTAPE IS SUFFICIENT FOR THE SIMULATION OF UP TO 2 MINI DISKS.

RM08.MAC HANDLES THE IOT'S FOR A DRUM TYPE RM08. IT IS ASSEMBLED ONLY IF DRM08(DEFINED IN P.MAC) IS 0.(DOESN'T RUN)

LINE.MAC HANDLES THE SIMULATION OF 680 LINES. ITS ASSEMBLY IS CONDITIONAL ON THE VALUE OF DCS680 IN P.MAC.(DOESN'T RUN)

PATCH.MAC IS PATCHING SPACE FOR THE SIMULATOR. MOST USERS WILL NOT FIND THEY NEED IT AND WON'T WANT TO LOAD IT WITH THE OTHER ROUTINES.

DRIVER.MAC IS THE FRONT END FOR THE PDP8 SIMULATOR. IT PROVIDES THE COMMAND DECODING ETC.

ASSEMBLING THE PDP8 SIMULATOR

THE STEPS FOR ASSEMBLING A SIMULATOR ARE AS FOLLOWS:

- A. PIP ALL FILES TO THE DISK
 ,R PIP
 DSK:/X/B-DTANI,*
- B. ASSEMBLE ALL OF THE FILES
 ,R MACRO
 SIM.CCL
- C. LOAD THE FILES TOGETHER
 ,R LOADER
 *PDP8,CORE,MEMORY,CONSOL,COMMON,TTY,
 *PTAPE,LPTSER,PLOTCH,DF32,DRIVERS
- D. SAVE THE LOADED PROGRAM.
 ,SAVE DSKISIM
- E. CREF LISTINGS CAN NOW BE OBTAINED BY:
 ,R CREF
 *LPT1:-A
 *LPT1:-B
 ETC. TO *LPT1:-N

CONDITIONAL ASSEMBLY PARAMETERS

THERE ARE A NUMBER OF CONDITIONAL ASSEMBLY PARAMETERS DEFINED IN P.MAC WHICH ARE:

A. VARIETY OF PDP8. THESE AFFECT THE OPERATE INSTRUCTION AND SIMULATED RUNTIMES.

A,PDP8 ;IF 0 A PDP8 IS SIMULATED
B,PDP8I ;IF 0 AN 8I IS SIMULATED
C,PDP8S ;IF 0 AN 8S IS SIMULATED

B. CORE UTILIZATION AND SIMULATOR SPEED WHICH ARE UNFORTUNATELY INVERSELY RELATED.

A,INLINE ;IF 0 IOT DISPATCHES AND MEMORY REFERENCES ARE FASTER BUT REQUIRES ANOTHER 1K OF PDP10 CORE TO RUN IN,
B,PACK ;IF 0 3PDP8 WORDS ARE PACKED INTO ONE PDP10 WORD, IF NOT 0 ONE PDP8 MEMORY WORD IS PLACED IN ONE PDP10 WORD. (PACKED TAKES AT LEAST 4US LONGER PER PDP8 MEMORY ACCESS, BUT TAKES 1/3 THE CORE)

C. IO DEVICE SIMULATION

A,PLOT8 ;IF 0, IOT'S ARE SIMULATED FOR A CALCOMP PLOT8,
B,UDF32 ;IF NOT 0 THE VALUE IS THE NUMBER OF MINI-DISKS TO BE SIMULATED. IF = 0 THE IOT'S ARE NOT SIMULATED

D. ACTION FOR ERROR CONDITIONS

A,ESTOP ;IF = 0, THEN 4 CONDITIONS WILL CAUSE AN AUTOMATIC TRAP TO THE DRIVER AFTER EXECUTION. THESE ARE:

1. EXECUTION OF AN UNRECOGNIZED IOT
 2. MICRO PROGRAMING RIGHT AND LEFT ROTATES IN A SINGLE INSTRUCTION.
 3. EAE TYPE INSTRUCTIONS.
 4. CYCLING IN MEORY, I.E. EXECUTING A NONJUMP INSTRUCTION IN LOCATION 7777.
- IF ESTOP=1, NO TRAP OCCURS AND IN ANY EVENT THE INSTRUCTION IS EXECUTED WITH THE PROPER EXECUTION TIME ACCUMULATED.

E. MEMORY SIZE.

A. MEMSIZ IS THE NUMBER OF WORDS IN THE PDP8'S MEMORY. I.E. +D4096, +08192, ETC.

MAINDECS

TO TEST THE SIMULATOR ALL OF THE FOLLOWING MAINDECS
HAVE BEEN RUN WHEN THE PROPER ASSEMBLY PARAMETERS WERE USED.

MAINDEC-81-D01B-PB INSTRUCTION TEST 1	3/25/68
MAINDEC-81-D02B-PB INSTRUCTION TEST 2	3/21/68
MAINDEC-08-D02B-PB INSTRUCTION TEST PART 2B	1/3/68
MAINDEC-08-D07B-PB RANDOM ISZ TEST	12/26/67
MAINDEC-08-D04B-PB RANDOM JMP-TEST	2/8/67
MAINDEC-08-D05B-PB RANDOM JMP-JMS TEST	12/28/67
MAINDEC-08-D06B-PB CALCOMP PLOTTER TEST	6/1/67
MAINDEC-08-D05DB-PB DF32 MULTI DISK	8/22/68
MAINDEC-08-D5CC-PB DF32 DISK DATA MINI DISK, INTERFACE, ADDRESS, DATA TEST	4/4/68

COMMENTS MAY BE DIRECTED TO:

DAVID MCCLURE
NORTHEAST REGIONAL OFFICE
DIGITAL EQUIPMENT CORP.
15 LUNDA STREET
WALTHAM, MASS

digital

INTEROFFICE MEMORANDUM

ERRATE SHEET DECUS NO. 6/10-43A

SUBJECT: 6/10-43a

DATE: March 17, 1971

TO: Angela Cossette, DECUS

FROM: Jim Lawson

DEPARTMENT: Washington, D. C.

One of my customers, National Security Agency (NSA), says he has found and solved two bugs in the PDP-8 simulator (6/10-43A). I took these patches over the phone and am not able to verify them.

1. When he tries to dump more than 2K, he gets a push down list overflow on PDL. PDL is in CONSOL. MAC.

FIX: doubled push down list from ten to twenty.

2. He says RDF and RIF instructions do not work correctly.

FIX: at RDF+1 change
 ANDI AC, 7
 to
 ANDCM I AC, 70

at RIF+1 change
 ANDI AC, 7
 to
 ANDCM I AC, 70



DECUS

PROGRAM LIBRARY

DECUS NO.	10-12
TITLE	PAL1Ø
AUTHOR	R. Bowering
COMPANY	Submitted by: N. Pappas Digital Equipment Corporation Maynard, Massachusetts
DATE	Submitted October 29, 1969
SOURCE LANGUAGE	MACRO-10

ATTENTION

This is a USER program. Other than requiring that it conform to submittal and review standards, no quality control has been imposed upon this program by DECUS.

The DECUS Program Library is a clearing house only; it does not generate or test programs. No warranty, express or implied, is made by the contributor, Digital Equipment Computer Users Society or Digital Equipment Corporation as to the accuracy or functioning of the program or related material, and no responsibility is assumed by these parties in connection therewith.

DECUS

YOUNG & RUBICAM



[The following text is extremely faint and largely illegible. It appears to be a multi-column layout, possibly a report or a letter, with several paragraphs and a vertical line separating columns. Some words like 'The', 'and', 'of', 'is', 'are' are visible but cannot be transcribed accurately.]

PAL10

DECUS Program Library Write-up

DECUS No. 10-12

ABSTRACT

PAL10 offers several advantages over to antiquated PALX assembler. Some of these improvements are:

- A. PAL10 runs in 4K as compared to 7K for PALX.
- B. Has a binary search on symbol table which significantly reduces processor time for any given assembly.
- C. Stacks symbol table listing in four vertical columns per page, reducing listing time and size.
- D. Has conditional assembly and literals.
- E. Two flavors of text handling.
- F. Assembler generated literals or links have been modified to flag the generated statement with a hash mark next to the object code and be counted as a link and not as an error.

PAL10 VERSION .134

PSEUDO	OP	CODES
*EXPUNGE		DELETE ALL SYMBOLS EXCEPT PSEUDO OPS
*FIXMRI		ARG-X000 DEFINE MEMORY REFERENCE
		INSTRUCTION HAVING VALUE "X000"
*FIXTAB		ESTABLISH ALL SYMBOLS CURRENTLY DEFINED AS
		ASSEMBLER SYMBOLS
*NOTE, PAL10		INITIALIZES ITS SYMBOL TABLE BEFORE EACH ASSEMBLY.
		I.E. FIXTAB, FIXMRI, AND EXPUNGE FUNCTIONS ARE VALID
		ONLY FOR THE ASSEMBLY IN WHICH THEY OCCUR
ZBLOCK	ARG	GENERATE BLOCK OF ZEROES
DECIMAL		CHANGE TO RADIX 10
OCTAL		CHANGE TO RADIX 8
DEFINE		DEFINE MACRO
*DTORG		GENERATE DECTAPE ORIGIN
DUBL	ARG	ACCEPT DOUBLE PRECISION
ENPUNCH		ENABLE BINARY OUTPUT
NOPUNCH		DISABLE BINARY OUTPUT
FIELD	ARG	GENERATE FIELD ORIGIN

1947

1947

The following information was obtained from the records of the Department of the Interior, Bureau of Land Management, for the year 1947. The information is presented in the form of a table, showing the number of acres of land owned by the United States, the number of acres of land owned by private individuals, and the number of acres of land owned by other entities. The information is presented in the form of a table, showing the number of acres of land owned by the United States, the number of acres of land owned by private individuals, and the number of acres of land owned by other entities.

CONDITIONAL	ASSEMBLY	PSEUDO OPS
IFDEF TAG < ARG>	ASSEMBLE	"ARG" IF "TAG" <u>IS</u> DEFINED.
IFNDEF TAG < ARG>	ASSEMBLE	"ARG" IF "TAG" IN <u>NOT</u> DEFINED.
IFZERO TAG < ARG>	ASSEMBLE	"ARG" IF "TAG" IS DEFINED AS <u>ZERO</u> .
IFNZRO TAG < ARG>	ASSEMBLE	"ARG" IF "TAG" IS DEFINED AS <u>NON-ZI</u>

PAGE Automatic origin at beginning of next core page

PAUSE NO-OPERATION

TEXT /ARG? Generate text table. "/" is the next delimiter. If number of characters is odd, table will be terminated with a 6 bit zero, if even table will be terminated by a 12 bit zero.

XLIST Toggle listing flip flop.

Literal Facility

[ARG] Define a location (determined by assembler) on page Ø which contains "ARG."

(ARG) Defines a location (determined by assembler) on current page which contains "ARG."

Operators

! Logical Shift 6-bits left

& Logical And

↑ Integer Multiply

<> Special, Alpha only, text mode when not used in Macro definitions or conditional assembly delimiters.

" Generate a word which contains the eight bit ASCII equivalent of the next character.

Error Indicators

C Illegal Character Error (this includes comment field)

D Illegal Redefinition

G Link Generated

1. The first part of the report deals with the general situation of the country and the progress of the work during the year. It is a summary of the work done and the results obtained. It is a general statement of the work done and the results obtained.

2. The second part of the report deals with the details of the work done. It is a detailed statement of the work done and the results obtained. It is a detailed statement of the work done and the results obtained.

3. The third part of the report deals with the details of the work done. It is a detailed statement of the work done and the results obtained. It is a detailed statement of the work done and the results obtained.

4. The fourth part of the report deals with the details of the work done. It is a detailed statement of the work done and the results obtained. It is a detailed statement of the work done and the results obtained.

5. The fifth part of the report deals with the details of the work done. It is a detailed statement of the work done and the results obtained. It is a detailed statement of the work done and the results obtained.

6. The sixth part of the report deals with the details of the work done. It is a detailed statement of the work done and the results obtained. It is a detailed statement of the work done and the results obtained.

- I Illegal Indirect
- L Literal error (overlap)
- M Macro Error
- N Number Error
- O OP code error
- P Phase or Multiply defined error
- Q Questionable format?
- T Illegal Text Character
- U Undefined Character
- Z Page Zero Literal or Exceeded Error

COMMAND STRING SWITCHES

- A Advanced Magtape One File
- B Back Space Magtape One File
- C Enable "CREF" Format (Switch must be set before listing device)
- D Enable DECTape Origins i. e. "DTORG"
- M Suppress Symbol Table Output
- N Suppress Errors on TTY
- T Skip to Logical End Of Tape
- W Rewind Magtape
- Z Zero DECTape Directory

In addition to the command string switches mentioned above, there are a group of special switches which are designed to tailor the assembler's symbol table to match up with other PAL assemblers. These switches modify only the symbol table including Pseudo-op codes. Operators including & and ! remain as their current function.

/P3	PAL III
/P8	MACRO 8
/PD	PALD
/PS	Super PAL or PAL 8

PAL10 General Notes

PAL10 does not currently possess a /X for core map.

Bug associated with Illegal use of equates has been fixed.

The passing of top of forms during XList CREF listings has been suppressed.

Bug associated with multi-device command strings has been fixed.

100
100
100

100

100

100

100

100

